

```

[SlaveLayout <layout_options>]
[SlaveServerOptions opt1 [val] ... optn [val]]
[SlaveScreenOptions opt1 [val] ... optn [val]]
5 [SlaveEnvironment var1=val ... varn=val]
[DefaultVisual
  [Depth <n>]
  [Class
    {PseudoColor|DirectColor|TrueColor|GrayScale}]
    [Layer {Image | Overlay}]
10 [Transparent]

[ScreenOptions opt1 [val] ... optn [val]]

```

Following the SLSD token is a list of *slave specifications*. The number of these specifications is dependent upon the slave layout. Finally, if an `<slsd_layout>` is specified, the Master will expect `<nRows> * <nCols>` *slave specifications*.

SlaveServerOptions is *optional* and defines server options that may be applied to all slaves in the system (including master's also behaving as slaves). For example, if all slaves need to have the DLEs load immediately, this mechanism may be used to prevent having to re-type information in the individual `<slave_spec>` ServerOptions entries. This may be entered as: SlaveServerOptions ImmediateLoadDles. The syntax of this option indicates that each option may or may not have a value. Options may be added on additional lines if necessary. For example:

```

SlaveServerOptions ImmediateLoadDles
                  HpCursorScaleFactor 2

```

SlaveScreenOptions is *optional* and defines screen options that may be applied to all slaves in the system. The syntax of this option indicates that each option may or may not have a value. Options may be added on additional lines if necessary. For example:

```

SlaveScreenOptions EnableIncludeInferiorsFix
                  HpCursorPriorityBoost 2

```

SlaveEnvironment is *optional* and defines one or more environment variables that may be set in the environment of all slave X servers (to also be inherited by the OGL Daemon). The syntax of this option indicates that each variable has a value. Environment variables may be added on additional lines if necessary. For example:

```
5      SlaveEnvironment      HPOGL_RENDER_FAST=1
                             HPOGL_DISPLAY_FRAMERRATE=1
```

DefaultVisual is *optional* and can be used to change the default visual for the entire SLS/d system. In previous installations of SLS/d, the default visual was selected by choosing the default visual of slave 0.

Depth is *optional* and specifies the default visual's depth. Typical values for <n> are 8 and 24.

Class is *optional* and specifies the default visual's visual class. One of the following values must be chosen: PseudoColor, DirectColor, TrueColor, or GrayScale.

Layer is *optional* and specifies whether the default visual shall live in the Overlays or in the Image Planes.

Transparent is *optional* and specifies that the default visual shall have a transparent entry in its default colormap.

ServerOptions is *optional* and defines server options that will only be visible to the Master. These screen options will not propagate to the slaves. If you want to use a big cursor, for example, this is where you would want to set the cursor scale variable (e.g., ServerOptions HpCursorScaleFactor 2).

ScreenOptions is *optional* and defines screen options that will only be visible to the Master. These screen options will not propagate to the slaves.

SLS/d Slave Layout (<layout_options>)

A SlaveLayout token may be used for specifying the SLS/d Slave Layout. For example,

```
5      slave_layout    ::= <slds_mode> | <slds_layout>

      slds_mode       ::= Mode [ Accelerate | Accumulate |
10     Supersample | Cave ]

      slds_layout      ::= [
                                Rows <nRows>
                                Columns <nCols>
15     ]
```

Therefore, to specify a non-Scalable 1x3 configuration, the user may enter:

```
      SLSd
      host1 host2 host3
20     SlaveLayout
          Rows 1
          Columns 3
```

To specify the Scalable Supersample Mode, the user may enter:

```
25     SLSd
      host1 host2 host3
30     SlaveLayout
          Mode Supersample
```

<slds_mode> and <slds_layout> are mutually exclusive. If both are specified, then the specification that appears last in the X*screens file will be used. In some cases, an error may be generated if the parser becomes sufficiently confused.

By way of illustration, FIG. 23 shows some possible configurations with their SLSd SlaveLayout lines. The **Accelerate Mode** example shows a 1x4 with a 2D slave (total of 5 Slaves). The Accelerate and Accumulate mode may be viewed as a plurality of 1x1's. **Supersample Mode** is actually a 2x2 SLS/d configuration, with an additional 2D Slave. Thus, in FIG. 23, the **Supersample Mode** example is shown as a 2x2 and the **Accelerate Mode** example is shown as a 1x4.